In the Claims

Please amend the claims as follows.

1. (Amended) A method of estimating results of a database query, the method comprising:

collecting workload information related to queries that have been executed on the database;

tracing query patterns of the queries in the workload to identify a usage of tuples in the database during execution of the queries wherein the usage of a given tuple relates to the frequency of access of a tuple ~~a relative frequency with which the given tuple was accessed by queries in the workload~~;

determining a sample weight based on tuple usage for each tuple;

performing a weighted sampling of the database based upon the sample weights; and

executing the database query on the weighted sample to estimate results of the database query.

2. (Previously presented) The method of claim 1 wherein the weighted sampling is performed by assigning a weight to each tuple based on a probability of usage of the tuple in executing the queries in the workload.

3. (Original) The method of claim 2 and further comprising computing an aggregate over values in each sample tuple.

4. (Original) The method of claim 3 wherein the aggregate is computed by multiplying each value by the inverse of the probability with which corresponding tuples were sampled.

5. (Amended) The method of claim 1 wherein the weights are a function of ~~the frequency of access of a tuple~~ and the number of queries in the workload that access the tuple.

6. (Original) The method of claim 1 wherein the tuple usage is stored on a page level.

7. (Amended) A machine readable medium having instructions for causing a machine to perform a method of estimating results of a database query, the method comprising:

collecting workload information related to queries that have been executed on the database;

tracing query patterns of the queries in the workload to identify a usage of tuples in the database during execution of the queries wherein the usage of a given tuple relates to the frequency of access of a tuple ~~a relative frequency with which the given tuple was accessed by queries in the workload~~;

determining a sample weights based on tuple usage for each tuple;

performing a weighted sampling of the database based upon the sample weights; and

executing the database query on the weighted sample to estimate results of the database query.

8. (Original) The machine readable medium of claim 7 wherein the weights are a function of the frequency of access of a tuple and the number of queries in the workload that access the tuple.

9. (Original) The method of claim 7 wherein the tuple usage is stored on a page level.

10. (Amended) A system that estimates results of a database query, the method comprising:

a module that collects workload information related to queries that have been executed on the database;

a module that traces query patterns of the queries in the workload to identify usage of tuples in the database during execution of the queries wherein the usage of a given tuple

relates to the frequency of access of a tuple ~~a relative frequency with which the given tuple was accessed by queries in the workload~~;

a module that determines a sample weights based on tuple usage for each tuple;

a module that performs a weighted sampling of the database based upon the sample weights ; and

a module that executes the database query on the weighted sample to estimate results of the database query.

11-34. (Canceled)

35. (Amended) A method of estimating results of a database and a given workload wherein the queries in the workload may have selection conditions, the method comprising:

collecting workload information related to queries that have been executed on the database;

tracing query patterns of the queries in the workload to identify a usage of tuples in the database during execution of the queries wherein the usage of a given tuple relates to the frequency of access of a tuple ~~a relative frequency with which the given tuple was accessed by queries in the workload~~;

determining a sample weight based on tuple usage for each tuple;

performing a weighted sampling of the database based upon the sample weights;

executing the database query on the weighted sample to estimate results of the database query, and,

generating a weighted outlier index.

36. (Original) The method of claim 35 and further comprising calculating an aggregate based on the samples of the index.

**REMARKS**

An abstract on a separate page has been included herewith in response to the Examiner's request. Claims 1-10 and 35-36 are pending in the subject application.

Claims 1,7, and 10 have been amended to eliminate the language objected to by the Examiner and to incorporate a feature previously recited in claim 5 into independent claims 1, 7, 10, and 35. Because this feature has been previously considered by the Examiner, the amendments to the claims do not raise issues requiring additional search or undue consideration. Accordingly, entry and consideration thereof is respectfully requested.

Claims 1, 7, 10, and 35 have been amended to address the § 112 rejection, in particular, the phrase "relative frequency with which the given tuple was accessed by the queries in the workload" has been replaced with the phrase "the frequency of access of a tuple" which was presented in original claim 5. Because this feature was recited in an original claim, the Applicant's representative submits that incorporation of this feature into the independent claims does not raise any § 112 issues.

Claim 1 stands rejected as anticipated by U.S. Patent No. 6,026,391 to Osborn. Claim 1 has been amended to recite a method of estimating results of a database query. Workload information is collected related to queries that have been executed on the database. Query patterns of the queries in the workload are traced to identify the a usage of tuples in the database during execution of the queries. The usage of a given tuple relates to the frequency of access of a tuple. A sample weight based on tuple usage is determined for each tuple. A weighted sampling of the database is performed based upon the sample weights. The database query is executed on the weighted sample to estimate results of the database query.

Osborn does not teach or suggest the method recited in claim 1. For example, Osborn teach or suggest assigning a weight to a tuple based on the tuple usage wherein the tuple usage is based on the frequency of access of the tuple or executing the database query on the sample to obtain estimate results of the query. With reference to claim 5, the Examiner stated in the present office action that Osborn does not teach tuple weights that are a function of the frequency of access of a tuple. In that office action, the Examiner cited Lohman as teaching this feature. The Applicant's representative respectfully disagrees. Lohman teaches weighting queries, not tuples, based on an expected frequency of the query. For these reasons, claim 1 and its depending claims 2 - 6 are in condition for allowance.

Independent claims 7 and 10 have been amended in a manner analagous to claim 1 and are in condition for allowance for the reasons stated above. Dependent claims 8-9 are, therefore, also in condition for allowance.

Claim 35 stands rejected as obvious in view of Osborn. Claim 35 has been amended to overcome the rejection with respect to Osborn and Lohman as discussed with respect to claim 1. Claims 35 and 36 are hence in condition for allowance.

In view of the foregoing discussion it is respectfully submitted that the claims of this application are patentably distinct from the prior art and each from the other and this application is in condition for allowance. Prompt notice to that effect is earnestly requested.

If necessary, the Commissioner is hereby authorized in this, concurrent, and future replies, to charge payment or credit any overpayment to Deposit Account No. 23-0630 for any additional fees required under 37 C.F.R. § 1.16 or 1.17; particularly, extension of time fees.

Respectfully submitted,

Date: 10 FEB 04

Jennifer Nock Hinton
Reg. No. 47,653

Telephone (216) 241-6700
Telefacsimile (216) 241-8151